UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/505,949	02/15/2000	Michael Chow	042390.P6447	5605

7590 01/25/2007

Thomas M Coester
Blakely Sokoloff Taylor and Zafman LLP
12400 Wilshire Boulevard Seventh Floor
Los Angeles, CA 90025

EXAMINER	
LI, AIMEE J	
ART UNIT	**PAPER NUMBER**
2183	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/25/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/505,949	CHOW ET AL.	
	Examiner	Art Unit	
	Aimee J. Li	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *02 November 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *09 January 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-19 have been considered. Claims 1, 10, and 18-19 have been amended as per Applicant's request.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE as filed 02 November 2006 and Amendment as filed 02 November 2006.

Claim Objections

3. Claim 19 is objected to because of the following informalities:

a. Claim 19 recites "performing a token specific operation when the processor is in at least a second word size ISA mode of the plurality of word size (ISA) modes." Please correct it to read --performing a token specific operation when the processor is in at least a second word size ISA mode of the plurality of word size [[(]]ISA[[)]] modes.-- The parentheses around "ISA" are unnecessary.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 10-17 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. As is, the current claim language has several interpretations and the scope of the claims is unclear.

6. For example, one reasonable interpretation of the claim is that determining what mode the processor is in, the first or second word size ISA mode, is not dependent on the checking

step. There is nothing in the current claim language relating the first and second word size ISA modes to the checking step, so the determination can be done by anything in the system and does not have to be dependent upon the token.

7. A second reasonable interpretation is that, to produce an output, the checking the mode step and the mode specific operations are not necessary. The method produces an output no matter what, and, with the conditional "if" steps that suggest the "if" steps are not necessary to produce an output in the claimed invention, any prior art that fetches input from a floating-point register, detects a token, and produces an output would read upon the claim.

8. A third reasonable interpretation is that producing an output is only possible when the processor is in the first word size ISA mode or in the second word size ISA mode, which was determined by using the checking step. The Examiner believes the third interpretation is the Applicants' intended interpretation for the claim language. Under this interpretation, the checking step is used to determine whether the processor is in a first or second word size ISA mode, operations happen dependent upon the mode, and an output is based upon the processor's mode. The Examiner suggests amending the language to "if the mode the processor is in is a first...if the mode the processor is in is a second...producing an output based upon the mode the processor is in", or similar, to clarify that what mode the processor is in is based upon the checking step and the output is based upon the processing mode.

9. The third reasonable interpretation is the interpretation the Examiner has applied for the rejections below.

Claim Rejections - 35 USC § 101

10. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

11. Claims 10-17 are rejected under 35 U.S.C. 101 because, under the third reasonable interpretation laid forth in the 112, second paragraph rejection above, when the processor is not in a first or second word size ISA mode, the final step would be detecting whether the input contains a token which lacks a tangible result as no usefulness of the token can be realized. The Examiner would note that claim 19 is not included in this rejection, because the claim language states that the processing step occurs "when the processor is in at least a first..." and the performing step occurs "when the processor is in at least a second...". The difference in the claim language is the use of "when", which implies that the condition will occur sometime, as opposed to "if", which implies that the condition is only a possibility that might or might not happen.

Claim Rejections - 35 USC § 102

12. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

13. Claims 1-5, 7-13, 15-17, and 19 rejected under 35 U.S.C. 102(b) as being taught by Blomgren et al., U.S. Patent Number 5,685,009 (herein referred to as '009), and U.S. Patent Number 5,781,750 (herein referred to as '750) incorporated by reference into '009 at column 3, lines 44-47. The shared registers disclosed in '009 are for use in the device disclosed in '750, as shown in '009 in column 1, line 24 to column 2, line 33; column 3, lines 444-67; and column 4, lines 6-16, hence '750 is incorporated by reference in '009 in the same embodiment. The

Examiner notes that claims 10-13 and 15-17 have been interpreted using the third reasonable interpretation laid forth in the 112, second paragraph rejection above.

14. Referring to claim 1, '009 and '750 have taught a processor comprising:

a. A first instruction set engine to process instructions from a first instruction set architecture (ISA), the first ISA having a first word size ('009 column 1, lines 24-33 and 44-48 and column 1, line 59 to column 2, line 4; and '750 Abstract; column 3, lines 51-55 and 59-65; column 6, lines 16-24; and Figure 2). In regards to '009 and '750, it is inherent that the two ISAs are different sizes, since RISC, specifically the PowerPC instruction set, and CISC, specifically the x86 instruction set, are different word sizes. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

b. A second instruction set engine to process instructions from a second ISA, the second ISA having a second word size, the second word size being different than the first word size ('009 column 1, lines 24-33 and 44-48 and column 1, line 59 to column 2, line 4; and '750 Abstract; column 3, lines 51-55 and 59-65; column 6, lines 16-24; and Figure 2). In regards to '009 and '750, it is inherent that the two ISAs are different sizes, since RISC, specifically the PowerPC instruction set, and CISC, specifically the x86 instruction set, are different word sizes. For more information please see the provided information for more information on the x86

and PowerPC instruction sets and Heuring and Jordan's Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

c. A mode identifier ('750 Abstract; column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; and Figure 2);

d. A plurality of floating-point registers shared by the first instruction set engine and the second instruction set engine ('009 column 2, lines 13-16; column 2, lines 41-58; column 10, lines 43-53; column 18, lines 3-16; column 19, lines 47-48; column 20, lines 10-15; Figures 6-8); and

e. A floating-point unit coupled to the floating-point registers, the floating-point unit processing an input responsive to the mode identifier to produce an output ('009 column 22, lines 48-54; and '750 column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; column 7, lines 1-12; and Figure 2).

15. Referring to claim 2, '009 and '750 have taught wherein the mode identifier is one of a plurality of bits in a processor status register ('750 Abstract; column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; and Figure 2).

16. Referring to claim 3, '009 and '750 have taught wherein the floating-point unit comprises:

a. Pre-processing hardware to detect if a token exists in the input ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2);

b. An arithmetic unit responsive to the input and the mode identifier ('009 column 1, lines 43-47; and '750 column 6, line 53 to column 7, line 12; and Figure 2); and

c. Post-processing hardware to perform a token specific operation if a token exists in the input ('009 column 3, lines 57-67; and '750 column 5, line 49 to column 6, line 13).

17. Referring to claim 4, '009 and '750 have taught wherein the input includes data stored in at least one of the floating-point registers ('009 column 4, lines 17-34; column 9, lines 21-28; column 18, lines 3-16; column 19, lines 47-48; column 20, lines 10-15; and Figures 7-8).

18. Referring to claim 5, '009 and '750 have taught wherein the input may contain a token ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2), wherein the floating-point registers are 82 bits wide ('009 column 2, lines 13-22), and wherein the token being an 82 bit processor known value ('009 column 1, lines 43-47 and column 2, lines 13-22 and 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2).

19. Referring to claim 7, '009 and '750 have taught wherein the floating point registers each comprise

a. A sign bit ('009 column 2, lines 12-22). In regard to '009, it is inherent that the significand, also known as the mantissa, includes the sign bit. Please see FOLDOC definition mantissa ©1996 provided with the Office Action dated 30 June 2003.

b. An exponent ('009 column 2, lines 12-22); and

c. A significand ('009 column 2, lines 12-22). In regards to '009, it is inherent and well-known in the art that a significand is the same as the mantissa. Please see David Goldberg's "What Every Computer Scientist Should Know About Floating-point Arithmetic" ©1991, specifically under the section titled "Floating-point Formats", paragraph 2 "where *d.dd...d* is called the *significand*" which refers to footnote 2 which says "This term was introduced by Forsythe and Moler [1967], and has generally replaced the older term *mantissa*."

20. Referring to claim 8, '009 and '750 have taught wherein the mode identifier indicates whether the processor is in a first mode or a second mode ('750 Abstract; column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; and Figure 2).

21. Referring to claim 9, '009 and '750 have taught wherein the mode identifier indicates whether the processor is in a 32 bit word ISA mode or a 64 bit word ISA mode ('009 column 1, lines 24-33 and 44-48 and column 1, line 59 to column 2, line 4; and '750 Abstract; column 3, lines 51-55 and 59-65; column 6, lines 16-24; and Figure 2). In regards to '009 and '750, the PowerPC instruction mode, i.e. RISC instruction mode, has 32 bit instruction words, as is the nature of the PowerPC instruction set. The x86 instruction mode, i.e. CISC instruction mode, has variable length instructions, which includes the 64-bit instruction length. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

22. Referring to claim 10, '009 and '750 have taught a method in a processor comprising:

a. Fetching an input from at least one of a plurality of floating-point registers ('009 column 2, lines 13-16; column 2, lines 41-58; column 10, lines 43-53; column 18, lines 3-16; column 19, lines 47-48; column 20, lines 10-15; Figures 6-8);

b. Detecting whether the input includes a token ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2);

c. If the token is detected in the input, checking what mode the processor is in ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2);

d. If the processor is in a first mode word size instruction set architecture (ISA) mode, processing the input to render an arithmetic result ('009 column 1, lines 43-47; and '750 column 6, line 53 to column 7, line 12; and Figure 2);

e. If the processor is in a second word size ISA mode, performing a token specific operation ('009 column 3, lines 57-67; and '750 column 5, line 49 to column 6, line 13); and

f. Producing an output ('009 column 1, lines 43-47 and column 3, lines 57-67; and '750 column 5, line 49 to column 6, line 13; column 6, line 53 to column 7, line 12; and Figure 2). In regards to '009 and '750, it is inherent that the two ISAs are different sizes, since RISC, specifically the PowerPC instruction set, and CISC, specifically the x86 instruction set, are different word sizes. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's Computer Systems

Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

23. Referring to claim 11, '009 and '750 has taught

a. Wherein the input is comprised of at least one operand and at least one operator ('750 column 3, lines 51-56). In regards to '750, the PowerPC RISC instruction set and x86 CISC instruction set both have at least one operand and at least one operator. Please see the provided information on the PowerPC and x86 instruction sets.

b. Wherein detecting comprises examining the at least one operand to determine whether any of the operands correspond to the token ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2); and

c. Wherein checking comprises examining a mode identifier to determine whether the processor is in the first mode or the second mode ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2).

24. Referring to claim 12, '009 and '750 have taught wherein processing comprises executing at least one operation on the at least one operand according to the at least one operator to achieve a result ('009 column 1, lines 43-47; and '750 column 6, line 53 to column 7, line 12; and Figure 2).

25. Referring to claim 13, '009 and '750 have taught wherein performing comprises propagating the token ('009 column 3, lines 57-67; and '750 column 5, line 49 to column 6, line

13); and wherein producing output comprises setting the output to be the token ('009 column 3, lines 57-67; and '750 column 5, line 49 to column 6, line 13). In regards to '009 and '750, the indication that the CPU is in emulation mode must be propagated through the entire process of switching from CISC to RISC and back to CISC.

26. Referring to claim 15, '009 and '750 have taught wherein checking comprises checking a mode identifier ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2).

27. Referring to claim 16, '009 and '750 have taught wherein checking comprises checking a mode identifier bit in a processor status register ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2).

28. Referring to claim 17, '009 and '750 have taught wherein the first mode is a 32 bit word ISA mode and the second mode is a 64 bit word ISA mode ('009 column 1, lines 24-33 and 44-48 and column 1, line 59 to column 2, line 4; and '750 Abstract; column 3, lines 51-55 and 59-65; column 6, lines 16-24; and Figure 2). In regards to '009 and '750, the PowerPC instruction mode, i.e. RISC instruction mode, has 32 bit instruction words, as is the nature of the PowerPC instruction set. The x86 instruction mode, i.e. CISC instruction mode, has variable length instructions, which includes the 64-bit instruction length. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

29. Referring to claim 19, '009 and '750 have taught a method in a multi-mode processor comprising:

a. Fetching an input from at least one of a plurality of floating-point registers ('009 column 2, lines 13-16; column 2, lines 41-58; column 10, lines 43-53; column 18, lines 3-16; column 19, lines 47-48; column 20, lines 10-15; Figures 6-8);

b. Detecting whether the input includes at least one token of a plurality of tokens ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2);

c. If at least one token is detected in the input, checking what mode the processor is in ('009 column 1, lines 43-47 and column 2, lines 51-58; and '750 column 3, line 65 to column 4, line 21; column 6, lines 53-59; column 7, lines 30-54; and Figure 2);

d. Processing the input to render an arithmetic result when the processor is in at least a first word size instruction set architecture (ISA) mode of a plurality of word size ISA modes ('009 column 1, lines 43-47; and '750 column 6, line 53 to column 7, line 12; and Figure 2); and

e. Performing a token specific operation when the processor is in at least a second word size ISA mode of the plurality of word size ISA modes ('009 column 3, lines 57-67; and '750 column 5, line 49 to column 6, line 13). In regards to '009 and '750, it is inherent that the two ISAs are different sizes, since RISC, specifically the PowerPC instruction set, and CISC, specifically the x86

instruction set, are different word sizes. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.

Claim Rejections - 35 USC § 103

30. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

31. Claims 6 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Blomgren et al., U.S. Patent Number 5,685,009 (herein referred to as '009), and U.S. Patent Number 5,781,750 (herein referred to as '750) incorporated by reference into '009 at column 3, lines 44-47 in view of InstantWeb's Online Computing Dictionary terms "speculative evaluation" and "speculative execution" (herein referred to as FOLDOC). The Examiner notes that claim 14 has been interpreted using the third reasonable interpretation laid forth in the 112, second paragraph rejection above.

32. '009 and '750 have taught wherein the token represents a "not a thing value" (NaTVal) that defines an unsuccessful load request ('750 column 4, lines 13-33 and column 7, lines 25-32). In regards to '750, the TLB is used to load the physical address of the instruction when virtual addresses are present and emulation mode is entered when a miss occurs in the TLB, i.e. cannot find the physical address to load, thereby creating an unsuccessful load request. FOLDOC has

taught speculative evaluation and execution of instructions (FOLDOC terms "speculative evaluation" and "speculative execution"). A person of ordinary skill in the art at the time the invention was made would have recognized that speculative evaluation and execution reduces the overall run-time of a process and keeps all functional units working, i.e. not wasted cycles, (FOLDOC terms "speculative evaluation" and "speculative execution"), thereby increasing processor speed and efficiency. Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the speculative evaluation and execution of FOLDOC in the device of '009 and '750 to improve processor speed and efficiency.

33. Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over Blomgren et al., U.S. Patent Number 5,685,009 (herein referred to as '009), and U.S. Patent Number 5,781,750 (herein referred to as '750) incorporated by reference into '009 at column 3, lines 44-47 in view of Amos Omondi's The Microarchitecture of Pipelined and Superscalar Computers ©1999 (herein referred to as Omondi). '009 and '750 have taught a multi-mode processor comprising:

a. A plurality of instruction set engines to process instructions from a plurality of instruction set architectures, the plurality of instruction set architecture each having a different word size ('009 column 1, lines 24-33 and 44-48 and column 1, line 59 to column 2, line 4; and '750 Abstract; column 3, lines 51-55 and 59-65; column 6, lines 16-24; and Figure 2). In regards to '009 and '750, it is inherent that the two ISAs are different sizes, since RISC, specifically the PowerPC instruction set, and CISC, specifically the x86 instruction set, are different word sizes. For more information please see the provided information for more information on the x86 and PowerPC instruction sets and Heuring and Jordan's

Computer Systems Design and Architecture provided with the action dated 18 June 2003 about the RISC and CISC instruction sets.;

b. A mode identifier ('750 Abstract; column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; and Figure 2);

c. A plurality of floating-point registers shared by the instruction set engines ('009 column 2, lines 13-16; column 2, lines 41-58; column 10, lines 43-53; column 18, lines 3-16; column 19, lines 47-48; column 20, lines 10-15; Figures 6-8); and

d. A floating-point unit coupled to the floating-point registers, the floating-point units processing an input responsive to the mode identifier ('009 column 22, lines 48-54; and '750 column 3, line 65 to column 4, line 2; column 4, lines 7-11; column 6, lines 53-57; column 7, lines 1-12; and Figure 2).

34. '009 and '750 have not explicitly taught a plurality of floating-point units. Omondi has taught a plurality of floating-point units (Omondi Figure 1.9; Figure 1.11; Figure 1.12; and Figure 1.13). A person of ordinary skill in the art at the time the invention was made would have recognized that a superscalar processor, such as those shown in Omondi, execute more than one instruction at a time, thereby increasing the speed and efficiency of a processor (Omondi page 6, section 1.3). Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the superscalar processor of Omondi in the device of '009 and '750 to increase the speed and efficiency of the processor. Also, merely duplicating a part of a device for multiple effect is not a patentable difference, see *In re Harza*, 274 F.2d 669, 124 USPQ 378 (CCPA 1960).

Response to Arguments

35. Applicant's arguments filed 02 November 2006 have been fully considered but they are not persuasive.

36. Applicant's have argued in essence on pages 6-9 and 12-17

> The Examiner's Answer cites Graf's Modern Dictionary of Electronics 6th Edition ©1984 ("Graf") to provide definitions for the terms "word" and "word size." Graf defines the term "word size" as "the number of decimals or binary bits comprising a word." As indicated by the Examiner, Graf defines the term "word" as "the number of bits needed to represent a computer instruction, or the number of bits needed to represent the largest data element normally processed by a computer." (See Graf, p. 1132)...
>
> ...
>
> .As indicated in the cited passages above, the "words size" of a processor is defined by the processor's ISA based on the assembly language instructions, which access data according to a defined size referred to as a word...

37. This has not been found persuasive. Applicant's have continuously argued the definition of "word size" throughout the prosecution of this application. The Examiner has previously stated, and cited support, that, in order for the definition argued by Applicants to be read into the claim, there must be an explicit, deliberate, and clear statement within the specification stating that "word size" means the argued definition. Barring such a statement, the Examiner is free to use any accepted definition of the term "word size" in the art. The Examiner has shown support for her use of the definition of "word size" being the number of bits representing an instruction in Graf page 1132. Applicant argues that this definition of "word size" in a processor is not

applicable and asserts that the only applicable definition of "word size" in a processor is the definition Applicant wishes to be read into the claim. However, it is unclear to the Examiner how the definition of "word size" she uses is not applicable. Applicant supports their arguments towards their definition of "word size" by citing page 1, lines 16-21 in the specification to show that their definition is explicitly stated in the specification. However, the cited passage merely states that the word size is defined by the ISA and is reflected by the data fetched from memory, which still falls in line with the Examiner's definition. Blomgren1 supports Examiner's stance in column 2, lines 12-22, which states that CISC, which typically has a wider instruction size, supports a wider data format for floating point registers than RISC, which typically has a smaller instruction size. To read that the "word size" as "the number of bits needed to represent the largest data element normally processed by a computer", as desired by Applicant's arguments, would be improperly limiting the claim language, since there is no specific indication within the claim language that the term "word size" is limited to this definition. Even if the cited passage from Applicants specification had a clear, concise, and, deliberate definition of "word size", without explicit language in the claim limiting the meaning of "word size", a person of ordinary skill would not recognize the meets and bounds of the claim language. Applicants argued definition is one of many definitions for the common term "word size", not a special term coined by Applicants that Applicants have defined clearly in the specification. The term "word size" is an accepted term in the art that has several accepted definitions, as shown in Graf, and to limit the definition based upon Applicants' arguments towards a specific definition would be improper and negligent by the Examiner. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant

relies (i.e., the definition of "word size") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

38. Applicants' argue in essence on page 10-17

> Applicants submit that although Huering and Jordan's Computer Systems Design and Architecture ("Heuring") may suggest that the instruction sizes may vary between CISC and RISC instructions, the cited passages of Blomgren1 and Blomgren2 teach that the instruction word size is the same for both the CISC and RISC sets, and in accordance with the definition of "word size" provide(*sic*) above...

39. This has not been found persuasive. Again, Applicants have continuously argued that Blomgren1 and Blomgren2 have taught instruction words of the same size, not different sizes. Applicants have chosen to focus on one paragraph within the entire disclosure of Blomgren1 and Blomgren2, which simply states that the instruction opcode has different sizes and locations for the opcode field in "the instruction word." Blomgren1 and Blomgren2 are referring to the specific instruction word that the opcode is found in, not that the two instruction sets have the same instruction word size. To insist that Blomgren1 and Blomgren2 teach that by this choice of wording, and the flexibility of English grammar of where modifying phrases can be placed in sentences (as the Examiner established in the Examiner's Answer dated 18 April 2006 with the Applicants' own claim language), is contrary to what is known in the art regarding CISC, RISC, PowerPC, and x86 instruction set architectures. The Examiner has continuously referred to Heuring as proof supporting her stance that RISC and CISC have different instruction sizes, but

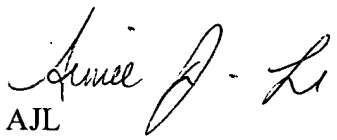
Applicants have chosen to argue that Blomgren1 and Blomgren2 have taught contrary to Heuring. Accordingly, the Examiner is providing yet further proof regarding her stance on RISC and CISC having different instruction sizes. Accompanying this Action, the Examiner provides additional proof that RISC and CISC, specifically PowerPC and x86 mentioned in Blomgren1 column 1, lines 34-42, have different instruction sizes. Roy Chartier's "Microprocessor Basics" from MONiTOR Magazine Online Volume 2, Issue 7 ©February 1995 states in Section RISC (Reduced Instruction Set Computing), paragraph 4 "RISC keeps the instruction size constant, where CISC instruction size, on the other hand, is greatly variable." In addition, Chartier uses the early x86 CISC architecture, e.g. Intel's 386 and 486 processor, and the PowerPC RISC architecture as examples of both RISC and CISC in section CRISC (Complex/Reduced Instruction Set Computing), paragraphs 6 and 10. Armin Gerritsen's "CISC vs. RISC" for the CPU Site ©March 1999 also describes the differences between RISC and CISC architectures and uses the x86 ISA as an example of CISC and the PowerPC as an example of RISC. Therefore, taking this further extrinsic evidence, and what has been known to those of ordinary skill in the art since the late 1970's, when RISC started becoming more popular within the industry (Chartier section RISC (Reduced Instruction Set Computing), paragraph 2), into consideration, the language "the instruction word" used in Blomgren is referring to the opcode's specific instruction, not the instruction size of the RISC and CISC architectures.

Conclusion

40. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J. Li whose telephone number is (571) 272-4169. The examiner can normally be reached on M-T 7:30am-5:00pm.

41. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

42. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



AJL
Aimee J. Li
20 January 2007